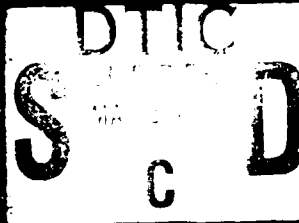
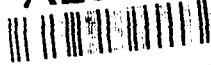


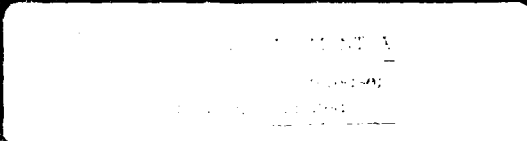
AD-A236 099



1



Quarterly Update



91-00417



91 5 23 024

Summary of SEI Accomplishments: 1Q91

This quarter, members of the Software Process Assessment Project and of the SEI Services group conducted the **second vendor assessment training class** for twenty-one people from five vendor organizations.

The **Software Metrics Definition Working Group** met for three days this quarter, and sub-groups were formed to discuss quality, size, and effort/schedule.

Rate Monotonic Analysis for Real-Time Systems project members presented rate monotonic analysis tutorials to various subcontractors on the BSY-2 program.

During the first quarter of 1991, modifications to **Serpent** included porting it to HP/UX for compatibility with the Army Common Hardware Software suite, continuing development of an interactive editor, and improving performance.

"Prospect for an Engineering Discipline of Software," published last year in *IEEE Software* and as technical report CMU/SEI-90-TR-20, was named one of the three best papers in *IEEE Software* in 1990.

The **Software Architecture Engineering Project** completed the first phase of work with the Air Force Electronic Combat Office (AFECO) to develop a standard electronic combat simulation architecture.

During this quarter, members of the **Domain Analysis Project** produced a tutorial on Software Reuse Technology at Tri-Ada and at the Ninth Annual Conference on Ada Technology.

In January, the **Software Risk Management Program** held a government workshop for Program Executive Officers (PEO) and senior-level executives from the services and the Office of the Secretary of Defense (OSD). Also during this quarter, the program completed its first assessment, which included assessment team training, a Quick Look assessment, and a detailed analysis of two risk areas.

The **Academic Series** began its eighth semester of production by videotaping three courses: Software Creation and Maintenance (version 2); Software Project Management (version 2); and Software Analysis (updates Software Verification and Validation).

In January, an **Education and Training Review Board** was established to provide quality assurance of educational materials.

A new three-day course for executives, **"Software Quality Improvement,"** was developed this quarter.

The **Computer Emergency Response Team Coordination Center (CERT/CC)** completed an initial outline of a computer security tutorial focused on Internet-connected UNIX systems.

New **industry affiliates** from ten organizations signed information exchange agreements during this quarter. Two **resident affiliates**, one from Texas Instruments and another from Army Communications-Electronics Command, joined the SEI.

.....
**This section provides
 a summary of
 accomplishments from
 January-March 1991**



Approval For	
SEC. 58241	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By Rec Form 50	
Distribution	
Availability Codes	
Dist	Avail and/or Special
A-1	

Table of Contents

Software Process Assessment	1	Software Process
Software Capability Evaluation	2	
Software Process Measurement	2	
Software Engineering Process Definition	3	
Software Acquisition Process Development	4	
Software Architecture Design Principles	5	Software Engineering Techniques
Software Architectures Engineering	6	
Software Development Environments	7	
Domain Analysis	8	
Domain-Specific Software Architectures	9	
CASE Technology	9	
Rate Monotonic Analysis for Real-Time Systems	11	Real-Time Distributed Systems
Real-Time Embedded Systems Testbed	13	
Software for Heterogeneous Machines	14	
User Interface	15	
Systems Fault Tolerance	16	
Ada 9X – Issues in Ada Adoption	16	
	19	Software Risk Management
Transition Methods	21	Special Projects
Empirical Methods	21	
Software Engineering Curriculum	23	Software Engineering Education
Master of Software Engineering	24	
Continuing Education.	24	
Binding of Ada and SQL	27	DARPA/STARS Support
CERT Coordination Center	29	CERT Coordination Center
	31	Affiliate Relations
	33	For More Information

Software Process

The Software Process Program focuses on improving the process of software development. Projects within the program are assessing the actual practice of software engineering in the defense community, training organizations to gain management control over their software development processes, supporting the use of quantitative methods and measures as a basis for process improvement, and developing improved methods for software process management.

The Software Process Assessment (SPA) Project helps organizations begin to improve their software development processes by providing a means of assessing their current practice. Software process assessment begins with an organization's formal commitment: by signing a Software Process Assessment Agreement and committing a team of software professionals to be trained in SEI assessment methodology. Objectives of the assessment are to understand the organization's current process of developing software; to identify key areas for improving the development process; and to help initiate process improvement.

In addition to providing both SEI-assisted and self-assessments, the project has recently launched a commercialization effort to allow trained, licensed vendors to provide authorized SEI assessments. This quarter, project members and members of SEI Services conducted the second vendor assessment training class. Twenty-one people from five vendor organizations (Contel, Dayton Aerospace Associates, American Management Systems, Digital Equipment Corporation, and Pragma Systems Corporation) attended.

Certified SEI personnel observed the first vendor-assisted assessments of the Software Productivity Consortium and Contel. Certified SEI personnel also coached self-assessments of the following organizations:

- 7th Communication Group, Pentagon
- Army Materiel Command, Fort Sill Systems Engineering, Oklahoma
- Boeing Aerospace, Seattle, Washington
- Motorola Management Information Systems/Network Computing System, Arizona
- Harris (Government Information Systems Division), Florida

In addition to the assessment activity, the project began planning the second state-of-the-practice report and the annual Software Engineering Process Group (SEPG) Workshop. This event will provide participants with practical knowledge about process improvement that participants can use at their own organization.

..... Software Process Assessment

.....

The Software Capability Evaluation (SCE) Project helps DoD acquisition organizations evaluate the capability of contractors to develop and maintain software competently. The project is improving and transitioning an evaluation method that examines the software process of contractors for use in software-intensive acquisitions.

This quarter, project members conducted two courses to train evaluation teams from the following organizations:

- Joint STARS Program
- Air Force Aeronautical Systems Command
- Naval Ocean Systems Center
- Army Missile Command
- MITRE
- Jet Propulsion Laboratory
- Electronic Systems Command

Project members also visited the Defense Systems Management College (DSMC) to present an overview of SCE to attendees of the week-long "Managing Software Acquisitions" course. While there, project members also discussed plans for transitioning SCE to DSMC.

Project members mailed the revised Capability Maturity Model (CMM) practices for level 2 to the CMM User Working Group for final review. These practices reflect the comments received from government, industry, and the Questionnaire Advisory Board on the June 1990 release of the CMM proposed baseline.

.....

The Software Process Measurement Project advocates the use of measurement in managing software development. The project is formulating reliable measures of the software development process and products to guide and evaluate development. To expedite DoD and industry acceptance, the project convenes a steering committee and two working groups, and undertakes a best-practices study, all devoted to encouraging organizations to use quantitative methods to improve their software processes.

During this quarter, the Software Metrics Definition Working Group met for three days, and subgroups were formed to discuss quality, size, and effort/schedule. The quality subgroup addressed comments by the Measurement Steering Committee about their draft on software problem reports. The size subgroup formulated rules and checklists for logical and physical source statements and continued work on assembling the software size metrics report. The effort/schedule subgroup worked on a method of decomposing the activities of a software development project into units of planned work that can be tracked; they also discussed characteristics that would be useful in comparing projects from an effort and schedule perspective.

The Software Acquisition Metrics Working Group met for three days in January to review the draft version of the Software Acquisitions document. This group is revising the document and anticipates holding a second peer review at its next meeting.

Project members participated in the Software Metrics Implementation Panel at San Antonio, the fifth software workshop sponsored by the Joint Logistics Commanders (JLC) Joint Policy Coordinating Group on Computer Resources Management (JPCG-CRM). Project members also presented a prototype of the findings from the Software Metrics Definition Working Group's size subgroup. This presentation was given at the REVIC (Revised Enhanced Version of Intermediate COCOMO) Workshop.

Project members visited Naval Air Development Center (NADC) to initiate a working relationship with them to improve their measurement capability. Members also worked with the Software Capability Evaluation Project to define measurement practices for the key process areas of the Capability Maturity Model.

The Software Engineering Process Definition Project supports process improvement through the maturation of the methods and technology associated with software engineering process definition. The project is developing the capabilities required to support the definition and evolution of software processes within organizations.

The project is supporting process development efforts at Standard Systems Center (SSC), Gunter Air Force Base, and Strategic Air Command (SAC), as well as exploring advanced applications of process through the Software Technology for Adaptable, Reliable Systems (STARS) Program.

During this quarter, the project held a two-day workshop entitled "Defining the Software Process - Getting Started." The workshop, which was held at SSC, covered basic principles of process management, process improvement, and process definition, as well as the descriptive modeling process, including group exercises that allow the participants to apply what they've learned.

The project provided on-site assistance to SSC in the following areas:

- a tactical plan for the Software Configuration Management Working Group
- a charter for the Management Oversight Working Group
- a project management bibliography for the Project Management Working Group

Members of the project met with representatives of SAC to review their process definition work and to provide assistance to their process improvement effort.

In addition to these activities, the project continued to provide support for the STARS Program.

Project members serve on the STARS Process Joint Activities Group. In this capacity, project members participated in the development of strategies, concepts, and plans that will guide and direct the process component of the STARS effort.

Project members also represented STARS at the Defense Advanced Research Projects Agency (DARPA) Software Process Workshop, and provided logistics support, planning, and coordination for this workshop.

●●●●●●●●●●●●●●●●
**Software Engineering
Process Definition**

Software Acquisition Process Development

Throughout the past quarter, project members have been developing a plan for implementing the first task, Bid Evaluation Methodology, and making contacts with DoD organizations that have software acquisition expertise.

Project members are planning to develop a methodology for evaluating the effort proposed by contractors to implement software (the Bid Evaluation Methodology). As the first step in the plan, project members have developed a draft of a survey questionnaire. Project members visited ESD to identify specific groups and people who could contribute to the upcoming survey. The survey is planned to occur during the second and third quarters; data analysis and a final report will be produced by the end of the year.

Software Engineering Techniques

The primary objective of the Software Engineering Techniques Program is to improve the practice of software engineering by improving individual and team productivity through the identification and transition to practice of emerging software technology. Promoting the appropriate use of this technology supports the SEI effort to transform software development from an ad-hoc, labor-intensive activity to a technology-supported engineering discipline.

To develop a fundamental understanding of structures for the software architecture level of design, the Software Architecture Design Principles Project is describing basic design elements used in the description, analysis, and development of software systems.

The task of identifying architectural constructs has progressed sufficiently to begin articulating this new area to potential recipients of models and tools. To this end, project members made the following presentations:

- MIT/Pew Foundation Workshop on Computer Science Curriculum: "Models for Undergraduate Courses in Software Engineering"
- Massachusetts Institute of Technology: "Larger-Scale Systems Require Higher-Level Abstractions"
- University of Massachusetts at Amherst: "Larger-Scale Systems Require Higher-Level Abstractions"

"Prospects for an Engineering Discipline of Software," published last year in *IEEE Software* and as technical report CMU/SEI-90-TR-20, was named one of the three best papers in *IEEE Software* in 1990.

"Models for Undergraduate Courses in Software Engineering," written by project leader Mary Shaw and James Tomayko, was accepted for publication in the proceedings of the Fifth SEI Conference on Software Engineering. The project leader also wrote and submitted for publication "Heterogeneous Design Idioms for Software Architecture."

Public service activities performed by the project leader this quarter include participation in the following:

- Information Science and Technology Study Group, DARPA Information Science and Technology Office (ISTO)
- Computer Science and Telecommunications Board of National Research Council

●●●●●●●●●●●●●●●● Software Architecture Design Principles

- **Board of Visitors, Computer Science Division, Office of Naval Research**
- **Panel for National Computer Systems Laboratory, Board on Assessment of National Institute of Standards and Technology (NIST) Programs, National Research Council**

The Software Architectures Engineering (SAE) Project has been involved in the engineering of software in several critical DoD application areas for the last five years. The project goal is to use sets of identified patterns together with engineering design goals to provide guidelines for building the basic elements of software structure (models). The project is currently focusing on three aspects of this technology: the generalization of software architecture by providing templated structural elements and connection rules that invest the implemented software with the desired product qualities; the naturalization of the models so that real-world components have corresponding software analogs; and the application of tools and notations to software development based on generalized structures.

handbook for a standard EC/TE system. SAE project members are serving as reviewers and are acting as a source for handbook material.

The SAE and DSSA Projects have submitted a proposal to Lieutenant Colonel Erik Mettala, manager of the DARPA DSSA Program, outlining SAE/DSSA project tasks in the continuing support of the DARPA effort.

SAE and DSSA project members have also begun working with NCSC personnel in Panama City, Florida, on the upgrade of the AN/SSQ-94 Combat System Integrated Trainer Equipment (CSITE). Together with a resident affiliate from NCSC who joined the SAE Project in December, project members are developing a course for training NCSC software developers in the adaptation and use of the Object-Connection-Update (OCU) structural model, to be delivered in April 1991. If the NCSC adopts the technology, the SAE and DSSA Projects will continue supporting the effort.

The Software Development Environments (SDE) Project focuses on environment support for software configuration management (SCM). In particular, the project has been assessing commercial advances in SCM concepts and functionality.

In addition to the checkout/checkin model, three SCM models have been observed: the composition model, the long transaction model, and the change set model. Each model is most appropriate for certain aspects of the software development process.

Configuration management capabilities can be found in SCM tools, CASE tools, and environment frameworks, each implementing its own variant of some of the concepts. This variety leads to the need to integrate tools with different SCM capabilities into a development environment, and the desire for a unified SCM model that can be adapted. As a result, the project has focused on the following activities:

- Creating a framework for relating SCM concepts found in actual SCM systems and assessing their impact on the software process.
- Providing an understanding of SCM as a key component of an SDE infrastructure.
- Addressing issues of integration between tools and SCM capabilities in environment frameworks with diverse SCM concepts.
- Exploring the feasibility of a unified SCM model and interface.

The project is communicating its findings on a framework for SCM concepts and the observed SCM models in several forums this quarter. First, project members prepared an all day tutorial entitled "State-of-the-Art in Environment Support for Configuration Management" for the 13th International Conference on Software Engineering (ICSE13). Second, the project leader is serving as the program chair for the 3rd International Software Configuration Management (SCM3) Workshop. Submissions have been reviewed by the program committee as well as project members, and the program has been put together. The paper "Concepts in Configuration Management Systems" by project member Susan Dart has been accepted as a keynote presentation at SCM3. Finally, the observed SCM models

●●●●●●●●●●●●●●●●
**Software Development
Environments**

have been discussed in a paper entitled "Configuration Management Models in Commercial Environments," which will be available as an SEI technical report.

The Next Generation Computer Resources (NGCR) Project Support Environment Standards Working Group (PSESWG) is an effort undertaken by the Navy to identify and select commercial standards in the environments area for use by the services. The first public meeting of the PSESWG was held during this quarter. The SEI is contributing to this effort in two ways. First, jointly with members of the CASE Technology Project, the results of a survey of experiences by project support environment builders were presented to the NGCR PSESWG executive committee and will be made available as an SEI technical report. Second, the project leader is a member of the PSESWG executive committee and an active participant of the reference model subgroup. The reference model subgroup is identifying and evolving a reference model as the basis for identification of interface areas in project support environments with potential for standardization.

.....
Domain Analysis

The Domain Analysis Project is developing and testing methods for performing domain analysis to support software reuse. The SEI has reviewed the Domain Analysis Project plan for 1991, and approved the plan subject to several minor changes. Project members are currently addressing issues raised during the review, including one about customers for domain analysis products. Project members met with staff members of the Future Battle Laboratory, Ft. Leavenworth, Kansas; they are interested in using the domain model of movement control as the basis for prototype development. The project will continue to work with this organization and Communications-Electronics Command (CECOM).

Project members are currently revising a report defining the context and scope of the Future Battle Laboratory domain analysis. This report will address the scope of the analysis and will describe the expected products and schedule. The report will also serve as an example for potential users of the domain analysis method.

The project has begun creating the domain model of the Army movement control domain. The model currently consists of a definition of the objects under movement and the data needed to precisely define and track the movement and operation and mission features of movement control.

The current plan is to have a draft domain model ready for review by CECOM and Ft. Leavenworth during the next quarter.

During this quarter, project members produced a tutorial on Software Reuse Technology at Tri-Ada and at the Ninth Annual National Conference on Ada Technology. Members of the project also completed a project briefing for the Army Strategic Defense Command Computer Resources Working Group. Also, project members produced an article entitled "Coming to Terms with Software Reuse Terminology: A Model-Based Approach," which will be published in a forthcoming issue of *ACM Software Engineering Notes*.

The Domain-Specific Software Architectures (DSSA) Project seeks to mature a technology base that supports all aspects of software development using structural models. Critical elements of this technology base are currently being pursued through collaborative relationships with external clients in the simulation and training application area. This foundation will serve both to extend the use of structural models within the simulator community and to establish new communities in other application areas.

.....
**Domain-Specific
Software
Architectures**

Both the C-17 and the B-2 aircrew training systems, two important users of DSSA's structural models, completed software development and began hardware-software integration (HSI). The Air Force program offices acquiring the training systems and the contractors developing them continue to be strong advocates of the technology.

Project members continued their efforts to accelerate maturation of the emerging structural modeling technology base through collaboration with two other users of structural modeling: the Special Operations Forces Aircrew Training System, and the BSY-2 Basic Operator Trainer. Focus is on the use of forms to facilitate both the specification and the mapping of requirements to structural model software templates on model-based documentation that realizes the intent, but not the extent, of MIL-STD-2167A, and on the use of model elements to improve resource estimation. Work also continued, with major contributions from Bill Schelker of the Aeronautical Systems Division/Deputy for Simulators, Director of Training Systems Development (ASD/YWB/ENETA), on the scope and quality of the structural model itself.

Future DSSA activities will address the institutionalization of the role of structural modeling in acquisition of training systems, and the creation of transition mechanisms such as handbooks and training courses.

The CASE (computer-aided software engineering) Technology Project was initiated in 1990. It focuses on improving the ability of SEI sponsors and affiliates to make informed decisions about tool adoption and to improve their practice in the use of CASE tools. It will also provide information to tool vendors on current tool usage and gaps in current technology.

.....
CASE Technology

Project members initiated a study of "lessons learned" from CASE adoption experiences. This effort analyzes how actual projects have used CASE tools. It examines initial expectations, actual experiences, and lessons learned from the successes and failures in using CASE tools. The effort considers the management and technical decisions made through the selection and adoption life cycle, together with the consequences of these decisions.

The technical report *CASE Tool Integration and Standardization* (CMU/SEI-90-TR-14) was published during this quarter. This report reviews the approaches to tool integration and summarizes both the major tool integration and standardization efforts along with their implications and potential.

In conjunction with the Software Development Environments Project, a study was conducted of the experiences of tool builders in integrating environments. This

effort assessed the use of current standards in the building of software engineering environments, and identified the need for additional standards.

Project members are designing a survey of the experiences of CASE tool users. The survey will address such issues as readiness for CASE tools, the adoption process, tool requirements and needs, and tool issues and problems. Project members are using information from the lessons learned study to design the survey.

The project conducts periodic workshops to consider critical issues in the CASE area and to provide a forum for vendors and users to address underlying issues in an objective setting. A workshop held in November 1990 focused on the CASE adoption process. The project is planning a second workshop for June 1991. This workshop will address several management issues in adopting CASE technology, including acquisition issues and selection issues.

◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇

**Software Engineering
Techniques
Reports**

January–March 1991

Spectrum of Functionality in Configuration Management Systems
(CMU/SEI-90-TR-11)

CASE Tool Integration and Standardization
(CMU/SEI-90-TR-14)

Transaction-Oriented Configuration Management: A Case Study
(CMU/SEI-90-TR-23)

Tool Version Management Technology: A Case Study
(CMU/SEI-90-TR-25)

Formal Development of Ada Programs Using Z and Anna: A Case Study
(CMU/SEI-91-TR-1)

Configuration Management Models in Commercial Environments
(CMU/SEI-91-TR-7)

Real-Time Distributed Systems

The goal of the Real-Time Distributed Systems Program is to improve the development of real-time distributed systems by integrating software engineering with systems engineering and reducing the risk of new technology.

The Rate Monotonic Analysis for Real-Time Systems Project is aiming to ensure that rate monotonic analysis and rate monotonic scheduling algorithms become part of the standard practice for designing, building, troubleshooting, and maintaining real-time systems. Rate monotonic analysis (RMA) allows engineers to understand and predict the behavior of real-time systems (both hardware and software) to a degree not previously possible.

• • • • •

Rate Monotonic Analysis for Real-Time Systems

Goals of the project include increasing the use of rate monotonic theory on highly visible projects, obtaining support for the rate monotonic approach from national hardware and software standards, and promoting the development of training and consulting resources outside the SEI. These resources are intended to serve as the major means for widely transitioning RMA techniques to real-time system developers.

As part of the goal to increase the use of rate monotonic theory on highly visible projects, project members are working closely with the prime contractor for the Navy's BSY-2 system, a major distributed system, to ensure the successful use of RMA. During this quarter, project members presented rate monotonic analysis tutorials to various subcontractors on the BSY-2 program. The project has developed and disseminated a rate monotonic data sheet to be completed for each BSY-2 computer software configuration item (CSCI). The sheets will help collect the data necessary to perform a rate monotonic analysis on individual CSCIs and then to perform an analysis of the integration of the CSCIs at design time. A similar sheet has been developed for describing the performance parameters of the executive software. Project members have also worked closely with software designers in performing rate monotonic analyses of selected CSCIs, and in all cases the analyses have helped to detect and rectify real-time design errors.

Project members are working with the Navy's Next Generation Computer Resources (NGCR) Program to encourage the development of a local area network (LAN) that adequately supports rate monotonic principles. Members have started to develop an Ada binding to the SAFENET Lightweight Application Service definition. The target implementation is the Xpress Transport Protocol developed by Protocol Engines Inc. (PEI). The Ada binding is being developed with the goal of obtaining predictable performance. The project is also interacting with PEI concerning the use of priorities in their hardware implementation of the protocol. Project members continue to work on the end-to-end scheduling issues in NGCR and

have raised several issues associated with the use of priorities in scheduling NGCR systems.

The NGCR has selected IEEE Futurebus+ as its computer backplane standard. During this quarter, project members continued to support the transition of Futurebus+ to the real-time community. Work in this area included reviewing chipset designs by Texas Instruments (TFB2010) for possible applications. Project members also worked with IBM on the real-time chapter of the *IEEE Futurebus+ System Configuration Manual*, which assists users of Futurebus+ in developing RMA-based applications. An overview of this chapter, "Real-Time Application Using IEEE Futurebus+," was accepted by *IEEE Micro* for publication. Project members are also examining the interface between Futurebus+ and other LAN standards. Some revisions to the IEEE Standard 896.3 have been made in response to comments from reviewers.

To ensure that the portable operating system interface standard (POSIX) provides adequate support for rate monotonic scheduling algorithms, project members have been working closely with the real-time POSIX working group. During this quarter, proposals regarding the priority inheritance protocols and the processor allocation scope for threads were incorporated into the standard; these proposals are being balloted now. If the ballot is approved, operating systems using the threads extensions to POSIX (1003.4a) will support the use of rate monotonic scheduling and its synchronization protocols, both for uniprocessors and shared memory multiprocessors.

The possibility of using the priority inheritance protocols under the POSIX real-time extensions (1003.4) was sought, and a new sporadic server proposal was presented. Both proposals support scheduling of aperiodic tasks under rate-monotonic scheduling (RMS) and will be discussed in the next working group meeting.

Project members reviewed proposals for the Ada 9X revision to ensure that the revised Ada standard also allows rate monotonic scheduling algorithms to be supported appropriately. The current version of the proposed changes provides strong support for RMA. Recommendations for improvements were made.

Project members have been interacting with the prime contractors of the Air Force's PAVE PACE program to ensure that they are aware of RMA and to encourage the use of rate monotonic principles when they offer system design proposals to the Air Force. The project is interested in PAVE PACE because the program is defining the next generation architecture for Air Force avionics systems. During this quarter, project members held technical meetings with each of the three contractors and discussed the applicability of RMA to some of their specific problems. Project members answered questions concerning RMA, and identified several areas where rate monotonic principles could potentially apply. In general, each contractor seemed more enthusiastic about RMA after the discussions.

Project members reviewed a NASA report that discussed a variety of sources of potential priority inversion in the design of Space Station Data Management System software. The report identified the sources and made appropriate recommendations for removing the problems. It also recommended that similar analyses be

undertaken for other subsystems. The report demonstrates the active use of RMA concepts on an important project in NASA.

Project members have been working toward the ultimate goal of producing an engineering handbook for applying rate monotonic analysis to real-time systems engineering. A draft of a report entitled *Rate Monotonic Analysis Adoption Rationale* has been completed as an intermediate step. This report is written in the spirit of the *Ada Adoption Handbook*, in a question and answer format. It captures many of the managerial and technical questions that are asked when organizations are first exposed to RMA. The report is currently being distributed for external review.

Project members are working with the Advanced Real-Time Technology Project at Carnegie Mellon University to extend the current analytical methods for assessing schedulability. Current methods assume for the most part that tasks execute with a fixed priority. The new methods will provide a more accurate assessment of schedulability when task priorities vary and will therefore allow a more accurate analysis of a wider class of real-time systems. In general, it allows the formalization of reasoning about tasks that do not abide by a rate monotonic priority assignment.

Project members have been successfully transitioning RMA to the research community, as well as to system developers. Research on extending and improving RMA is underway at Texas A&M (in support of the Space Station Freedom), at the University of Massachusetts (which produced a technical report proposing a modified rate monotonic scheduling algorithm for use in distributed systems), and in thesis work being undertaken at Florida State University.

Project members have begun transitioning the sporadic server to the real-time community through contacts with different Ada vendors and potential users. The sporadic server is the scheduling mechanism introduced by rate monotonic theory to provide enhanced schedulability and analyzability for aperiodic tasks. The sporadic server algorithm has many interesting applications for real-time systems; however, no commercial operating systems or Ada runtime systems currently implement this algorithm.

Project members have developed a solution for users who want to begin using the sporadic server while waiting for commercial runtime implementations to appear. This interim solution is a modification of the sporadic server algorithm that can be implemented as an application-level task in Ada or with any preemptive operating system, provided that there is a user-callable mechanism for dynamically changing the task priorities. Although this sporadic server implementation has more overhead than a runtime or kernel implementation, the worst and average case performances are comparable in most cases.

The Real-Time Embedded Systems Testbed (REST) Project is collecting, classifying, generating, and disseminating information about Ada performance issues in real-time embedded systems.

This quarter, the project continued to refine the detailed design of the Hard Real-Time Benchmark Suite (Hartstone). The first benchmark of the suite, the periodic

●●●●●●●●●●●●●●●●
**Real-Time Embedded
Systems Testbed**

harmonic (PH) tests, is now being used by nearly 300 sites. The European Space Research and Technology Center (ESTEC), an organization of the European Space Agency, has used Hartstone to evaluate Ada compilers for space applications. Several vendors are now routinely using Hartstone to evaluate their products, and some C versions of Hartstone that use the UNIX process model have appeared.

In support of BSY-2, the project investigated the performance impact of Ada style using Ada Compiler Evaluation Capability (ACEC), Ada Evaluation System (AES), Performance Issues Working Group (PIWG), and Hartstone benchmarks. Feedback was offered to the providers of Ada compiler evaluation software about their products. Project members participated in monthly meetings with General Electric and the compiler vendor (Verdix) to continue improving the evaluation technology, the compilation system, and the performance of the BSY-2 software.

Project members began informal collaboration with the Comet Rendezvous Asteroid Flyby (CRAF) Cassini mission of NASA's Jet Propulsion Laboratory (JPL). They are following the progress of and offering occasional advice to the mission's development team, which is designing a new generation of spacecraft. The spacecraft are scheduled to study Saturn and its moons (Cassini mission) and the smaller bodies of the solar system (CRAF mission). The tremendous radiation on Jupiter and Saturn requires MIL-STD-1750A hardware architecture; the flight software will be developed in Ada. Project members are organizing a workshop about MIL-STD-1750A and Ada to be held during the summer at JPL.

REST project members are also working cooperatively with two other SEI projects. Together with the Rate Monotonic Analysis for Real-Time Systems (RMARTS) Project, REST investigated classes of benchmarks that would produce the parameter values necessary to perform an accurate rate monotonic analysis of the schedulability of Ada tasks. Project members also started to evaluate the overhead of sporadic servers. With the Software for Heterogeneous Machines Project and their resident affiliate from Hughes Aircraft Company, REST started to investigate how Specification Methodology for Adaptive Real-Time Systems (SMARTS) applies to the automatic generation of application-specific benchmarks.

●●●●●●●●●●●●●●●●
**Software for
Heterogeneous
Machines**

The Software for Heterogeneous Machines Project is developing tools and a methodology for building distributed, large-grained, concurrent applications to run on networks of heterogeneous machines. The project has developed Durra, a language for describing distributed applications as a set of task descriptions and for writing type declarations that prescribe a way to manage the resources of a network.

In conjunction with the Institute for Simulation and Training (IST) at the University of Central Florida, project members are using Durra to develop emulators for networks of training devices. Members of the project completed the third quarterly report for this work, which is sponsored by the Army Program Manager, Training Devices (PM-TRADE). The project leader briefed PM-TRADE and DARPA on work that is being done for IST. The project leader also discussed with Major James Wargo (DARPA) potential follow-up work. Major Wargo expressed interest

in visiting the SEI in the near future to discuss broad range interoperability of simulators.

Also during this quarter, project members refined the Durra language and began upgrading the compiler and reimplementing the runtime support. Instead of generating each task to run as a separate program, the new compiler will generate multiple clusters of tasks linked together to run as a single program. The new runtime system will eliminate most of the performance penalty due to network communications overhead. These extensions should apply directly to other work being done for IST.

Project members drafted a statement of work for a potential project to be sponsored by the U.S. Army Communications-Electronics Command (CECOM) Center for Software Engineering. The work would entail developing a testbed for experimenting with real-time scheduling policies and proposed extensions to Ada that CECOM is sponsoring at the Illinois Institute of Technology. Although CECOM cannot support this activity at this time because of limited funds, CECOM wants to facilitate cooperation between the SEI and their contractors.

In addition, a member of the project attended the Real-Time/Reuse Technical Interchange Meeting sponsored by the CECOM Center for Software Engineering. The meeting provided a forum for members of the real-time and software reuse communities to exchange information and discuss issues of interest to CECOM.

The User Interface Project is developing Serpent, a user interface management system (UIMS). Serpent separates the concerns of the user interface from those of the application, which allows integration of input/output (I/O) technologies without modification of the functional portion of the application.

.....
User Interface

Serpent is distributed electronically through anonymous file transfer protocol (FTP) and is supported electronically through a mailing list. Currently, more than 150 Serpent sites from research, academia, and industry are on the mailing list.

During the first quarter of 1991, modifications to Serpent included porting it to HP/UX for compatibility with the Army Common Hardware Software suite, continuing development of an interactive editor, and improving performance. The documentation is also being completely revised: drafts of four manuals have been completed and are currently being edited; three drafts are nearing completion.

Project members presented Serpent in an Application Builders Tutorial at the X Technical Conference in Boston. The tutorial was attended by more than 200 people.

The User Interface Developers Workshop, organized by the project, held two meetings this quarter. A white paper that presents a reference model for future user interface tools is currently in production and is scheduled to be completed by May 1.

The project leader attended several review meetings during this quarter at which language revision proposals were presented to the Ada 9X Distinguished Reviewers (a group established by the Ada 9X program manager to review all aspects of the Ada 9X effort). The project leader also attended a workshop at which the revision proposals were presented to the public.

Generic Avionics Software Specification
(CMU/SEI-90-TR-8)

Rate Monotonic Analysis for Real-Time Systems
(CMU/SEI-91-TR-6)

◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇

**Real-Time Distributed
Systems Reports**

January–March 1991

Software Risk Management

The goal of the Software Risk Management Program is to improve the management of risks that arise in the development of software-intensive systems. In this context, "risk" refers to the uncertainty and impact associated with an event; "management" refers to the identification and resolution of the risk. Managing risk, therefore, entails identifying those things that can go wrong and assessing their likelihood and impact. A premise of the Risk Program is that confronting risk in a systematic way is fundamental to controlling the quality, cost, and schedule of software products.

The Risk Program is developing a framework for managing software risks at all levels during the entire software life cycle. The foundation of this framework provides the underlying principles of, and a paradigm for, risk management. Through workshops and field interviews, the program is collecting data about the current practice of risk management and assessing the needs of the community. The initial focus is on what are probably the least understood and least disciplined activities in software acquisition and development: software risk identification and analysis.



The Risk Program is continuing to conduct interviews with industry to gather information on the current practice of risk management and risk communication. The program is initiating an effort to develop a questionnaire for the government that will enable the gathering of information on how the services handle risk management and risk communication. The program is planning to start conducting government interviews in the third quarter of 1991.

The Risk Program held a government workshop for Program Executive Officers (PEO) and senior-level executives from the services and the Office of the Secretary of Defense (OSD). The workshop was held at Defense Systems Management College (DSMC) on January 15 and 16. The objective of the workshop was to identify the issues and needs regarding risk management within the services and for large-scale software intensive systems in particular. The workshop produced a prioritized list of needs from the PEO perspective. DSMC has requested that the Risk Program conduct another workshop for program managers in the second quarter of 1991.

As part of the framework for managing software risks, the Risk Program has been concentrating on risk identification. The risk identification mechanisms comprise the first step in the risk management process. To date, the program has developed the following techniques for risk identification:

- Risk Appraisal
- Taxonomy-Based Questionnaire
- Matrix Mechanism

The Risk Program is currently testing the utility of these mechanisms as part of the risk assessment process.

The program has developed and tested a two-step process for risk assessments. The first step provides a quick, broad-brush view of risk areas from the system perspective. The second step provides a detailed analysis of each risk area to identify specific risks that may be acted upon. The process allows for prioritizing specific software risks in the context of the overall system development priorities.

All risk assessments are confidential and are not attributable to individuals. Also, a joint team (SEI participants and program participants) is trained in the process and in applying specific risk identification mechanisms.

This quarter, the program completed its first assessment, which included assessment team training, a Quick Look assessment, and a detailed analysis of two risk areas. The program received commitment from another organization to conduct its second assessment and is preparing for the team training.

Special Projects

The Transition Models Project will evolve a set of methods for planning, implementing, and assessing transition activities that will be useful for technology producers and consumers both inside and outside the SEI. Transition Models staff will also provide SEI staff with education and training on technology transition concepts and approaches.

..... Transition Models

Priscilla Fowler, leader of the Transition Models Project, is coordinating SEI support and providing input on technology transition to the DoD Software Technology Plan development effort. In this quarter, she attended planning meetings and working sessions, prepared a draft of the technology transition section of the DoD Software Technology Plan, and submitted the section to the coordinators of the plan at the Office of the Secretary of Defense (OSD).

Project members and John Maher, another SEI member of the technical staff, have developed a tutorial on software technology transition. The tutorial includes material on managing technology maturation and adoption, managing technology innovation in an organizational context, and developing an organizational strategy for technology transition. The tutorial will be presented at the 13th International Conference on Software Engineering, to be held in May in Austin, Texas.

In February, Priscilla Fowler and Stan Przybylinski presented part of the tutorial to the Software Technology for Adaptable, Reliable Systems (STARS) Program architects in Washington DC. Fowler and Przybylinski, with Judy Bamberger of the SEI STARS Support Project, presented their analysis of STARS technology transition planning to date, and participated in extensive discussions on how to proceed with technology transition planning for STARS.

Fowler and Przybylinski also presented a subset of materials from the tutorial to STARS program managers, service representatives, and support contract personnel at UNISYS in Reston, Virginia, as part of the STARS PMR (program management review) meeting held in February.

During this quarter, the project began reviewing several SEI projects as candidates for testing transition frameworks.

The Empirical Methods group supports technology development, assessment efforts, and SEI technology projects by providing market research methods and materials, conducting surveys, and evaluating events or validating products of SEI projects. In addition, the group conducts the National Software Capacity Study.

..... Empirical Methods

In the first phase of the study, 106 industry and Air Force executives were asked to rate the relative importance of factors that contribute to failure to meet schedules and budgets for software development contracts.

Empirical Methods staff also reported on readily available data about demand for Ada software, growth in demand for post-deployment software support (PDSS), and labor supply for software and systems development.

Work on the second phase of the study continued this quarter. In the second phase of the study, Empirical Methods staff and colleagues from Carnegie Mellon's School of Urban and Public Affairs are undertaking three large-scale data collection efforts that will continue for three years. First, data are being obtained to estimate the labor supply and career patterns of scientists and engineers involved in software production in the U.S. Civil Service. Second, data collection began to estimate labor requirements to do PDSS work and to analyze who is doing PDSS work for the Air Force and the Army. Third, data are being collected to update and improve the accuracy of the initial estimates of demand for Ada software production found in *National Software Capacity: Near-Term Study*. All the material resulting from data collection and analysis will be incorporated in future capacity study briefings and reports.



Special Projects Reports

January–March 1991

National Software Capacity: Near-Term Study
(CMU/SEI-90-TR-12)

National Software Capacity: Near-Term Study Executive Summary
(CMU/SEI-90-SR-12)

Software Engineering Education

The primary objective of the Software Engineering Education Program is to increase the number of highly qualified software engineers by rapidly improving software engineering education throughout the education communities of academia, government, and industry. To accomplish this, the projects of the Education Program focus on accelerating the development of software engineering programs in academia and on enhancing opportunities for the continuing education of practitioners.

The Software Engineering Curriculum Project is developing model curricula, promoting the growth of graduate software engineering programs in the academic community, investigating the feasibility of undergraduate programs, and working to increase the amount of software engineering content in both undergraduate and graduate computer science programs. The project produces educational materials, including the Academic Series, a set of videotaped graduate-level courses on software engineering.

The SEI presented four 90-minute tutorials at the 22nd SIGCSE Technical Symposium on Computer Science Education. Held in San Antonio, Texas, this symposium was sponsored by the Association for Computing Machinery's Special Interest Group for Computer Science Education. Tutorial topics were: Software Quality Issues, Software Technical Reviews, Unit Testing and Analysis, and User Interface Development. The first two topics were presented by SEI staff; the latter two were presented by university educators who developed materials as SEI visiting scientists.

In January, the Academic Series began its eighth semester of production by videotaping three courses: Software Creation and Maintenance (version 2); Software Project Management (version 2); and Software Analysis (updates Software Verification and Validation). Presentation of these courses began at Carnegie Mellon in January. The videotaped versions will be available to participating schools for the fall 1991 semester.

Among the institutions presenting Academic Series courses during the spring semester are Florida Atlantic University (six sites), Regis College, Rose-Hulman Institute of Technology, and Virginia Polytechnic Institute and State University. Tutors from these institutions attended a one-day orientation session at the SEI.

Responsibility for the Academic Affiliates program was recently transferred to the Program Development Division. This new arrangement enables the SEI to interact with all its affiliates—academic, industry, and government—through one focal point.

●●●●●●●●●●●●●●●●
**Software Engineering
Curriculum**

.....

Master of Software Engineering

In response to industry's growing demand for skilled software developers, Carnegie Mellon now offers a 16-month master's degree program in software engineering (MSE). The program is a joint effort of the Carnegie Mellon School of Computer Science and the SEI. The core of the program is based on the SEI curriculum recommendations for MSE programs.

In the spring 1991 semester, Carnegie Mellon offered four courses in the MSE program: Software Analysis, Software Creation and Maintenance, Software Project Management, and Formal Design. These courses were developed in conjunction with the Software Engineering Curriculum Project.

The Software Development Studio, an ongoing project course, continued through the spring semester. The students are currently finishing the specification of requirements and drafting a project management plan.

.....

Continuing Education

The Continuing Education Project interacts with industry and government to increase the availability of high-quality educational opportunities in software engineering topics for software practitioners and executives. The project produces the Continuing Education Series and the Technology Series. In the Continuing Education Series are video-based courses designed for clients' in-house education and executive offerings designed for decision makers involved in improvement efforts. The Technology Series provides stand-alone presentations that promote awareness of emerging issues and leading-edge technologies.

In January, an Education and Training Review Board was established to provide quality assurance of educational materials. The board provides guidance for course developers throughout the life cycle of instructional development, and is responsible for approving course materials before their release as SEI products. The board is administered through the Products and Services Division and is chaired by the Continuing Education project leader.

Software Project Management for Instructors was offered March 4-15. The class of 15 included instructors from Grumman Melbourne, Honeywell, Logicon, MITRE, Naval Electronic Systems Engineering Center, Naval Ocean Systems Center, and Scott Air Force Base. One attendee was an SEI resident affiliate from U.S. Army Communications-Electronics Command (CECOM). In addition to instruction on software project management, the attendees received guidance in tailoring the course to meet the educational requirements of their organizations. They are now qualified to serve as learning facilitators for the course at their organizations. Upon completing the course, participants received 5.4 continuing education units from Carnegie Mellon.

A one-day version of "Software: Profit Through Process Improvement" was presented to 34 SEI employees on March 21. The full-length version is regularly offered as a 2-day course for executives.

A new course for executives, "Software Quality Improvement," was developed this winter. In this 3-day course, executives begin preparing action plans tailored to the needs of their organizations. The course consists of 12 tutorial presentations and 9 hands-on planning sessions. Software Quality Improvement was offered for

the first time on March 25-27. The class of 18 included executives from Control Data Corporation, General Dynamics, Grumman, GTE, IBM, Logicon, Space/Naval Warfare Systems Command, U.S. Army CECOM, and Union Switch and Signal.

A new videotape, "Executive Leadership for Software," was added to the Technology Series. In this videotape, Watts Humphrey discusses how managers can make a difference in efforts to improve our national software capability. Humphrey also describes an improvement strategy based on the capability maturity model developed at the SEI.

Project staff made two significant presentations this quarter: "Addressing the Shortfall of Software Professionals" (presented at the executive session of Air Force Bold Stroke at Maxwell Air Force Base); and "University-Industry Collaboration: Educational Opportunities" (presented at the Center for Advanced Technology in Computer Applications and Software Engineering, Syracuse University).

DARPA/STARS Support

As Ada use becomes more common in software-dependent systems, and as requirements grow, system developers will be faced with both the benefits and problems of adopting more disciplined and integrated approaches to software engineering. The goals of DARPA/STARS (Software Technology for Adaptable, Reliable Systems) Support are to serve as a focal point and facilitator in removing technical and managerial impediments to the adoption of Ada; to support the STARS Program in technology development, integration, and transition efforts; and to support DARPA efforts to develop, evaluate, and disseminate new software engineering approaches derived from the development of software architectures characteristic of particular application areas.

Two commercial implementations of the SAMEDL (SQL Ada Module Description Language) are under development: Intermetrics of Cambridge, MA and CCI of Meppen, Germany. This is in addition to the announcement of support for the SAME methodology made last year by Informix.

A technical report, *A Rationale for the SAMeDL* (CMU/SEI-91-TR-4), is in production. This document is a companion to the SAMeDL reference manual (*The SQL Ada Module Description Language SAMeDL*, CMU/SEI-90-TR-26), produced last year. Whereas the reference manual gives the precise syntax and semantics of the SAMeDL, the rationale document gives the design principles behind the SAMeDL and explains how the language is intended to be used.

STARS/Users Workshop: Final Report – Issues for Discussion Groups
(CMU/SEI-90-TR-32)

Rationale for SQL Ada Module Language Description (SAMeDL)
(CMU/SEI-91-TR-4)

● ● ● ● ● ● ● ● ● ● ● ● ● ●

Binding of Ada and SQL

◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆

**DARPA/STARS
Support
Reports**

January–March 1991

CERT Coordination Center

The Computer Emergency Response Team Coordination Center (CERT/CC) supplements existing mechanisms by which informally organized experts deal with and prevent computer emergencies. The CERT/CC at the SEI supports two different communities: Internet users, and developers of technology that is available on the network, such as UNIX and networking software. The CERT/CC provides a dependable 24-hour point of contact for security issues and allows rapid communication during emergencies. It also raises constituents' awareness of security issues and assists individual organizations in improving the security of their systems. Finally, the CERT/CC maintains a highly secure repository of information for team members and cultivates close ties with researchers in the area of trusted systems to improve the security of existing systems.

Since its inception in 1988, the CERT/CC has responded to a continuous stream of reported security incidents. These include reports of intrusions, worms, and viruses, as well as reports of vulnerabilities and suggested fixes for problems. In handling these problems, the CERT/CC issues advisories to the Internet community to warn them of problems and inform them of preventive techniques. In cases where vulnerabilities exist, the CERT/CC works with software vendors and the technical community in analyzing and resolving the problems.

During the first quarter of 1991, the CERT/CC continued working with the Internet Engineering Task Force (IETF) to produce recommendations for Internet security policy. The Security Policy Working Group met to resolve issues and to produce final recommendations, which have been forwarded to the Internet Advisory Board for approval and further action.

.....

CERT members participated in the Site Security Policy Handbook Working Group, a second IETF working group whose aim is to produce a security handbook for use by Internet-connected site and system administrators. Ongoing meetings of the working group have resulted in a first draft of the handbook, which outlines key issues and provides guidance on policy, administrative, and technical issues to support administrators efforts at improving the security of their systems. Final editing of the handbook is scheduled for May 1991, when it will be released as an IETF RFC (request for comments) draft.

An initial outline of a computer security tutorial, focused on Internet-connected UNIX systems, was completed. Current plans call for completing and piloting the tutorial by August 1991.

CERT/CC staff members hosted working meetings and participated in professional conferences. The focus of these activities is to raise awareness of security issues and to support organizations' efforts at improving the security of their operational systems.

The CERT/CC continued to take a lead position in the development of the CERT System, a federation of organizations working together to improve the security of their systems. CERT System charter members, 11 federally funded groups, have formed a steering committee (chaired by a CERT/CC staff member) and elected a Secretariat (a representative from the National Institute of Standards and Technology). They have also formed three working groups focused on: inter-CERT communications mechanisms, information content and format standards, and workshop and conference agendas. A CERT System workshop was held in March 1991; details of backup communication mechanisms, advisory content and format, and future workshops were discussed. The agenda and format of the third annual Computer Security Incident Response Workshop, scheduled for August 1991, was finalized.

Affiliate Relations

The Affiliate Relations Function establishes and maintains SEI relationships with industry, government, and academia, providing access to SEI information through mailings, telephone contact, special meetings, and symposia. Affiliate Relations also cooperates with the SEI Joint Program Office to negotiate for and place resident affiliates at the SEI.

New industry affiliates from 10 organizations signed information exchange agreements during this quarter: AGS Management Systems, Inc.; Atherton Technology; Computing Trends; David Maibor Associates, Inc.; Electronic Data Systems, Corporate Technical Architecture Division and Technical Systems Development Division; I-NET, Inc.; J.G. Van Dyke & Associates, Inc.; KPMG Peat Marwick; and OAO Corporation.

In this quarter, 3 resident affiliates concluded their work at the SEI. New resident affiliates, one from Texas Instruments and another from Army Communications-Electronics Command, joined the SEI. As of March 31, 1991, 13 resident affiliates were working at the SEI: 6 from industry, 1 from academia, and 6 from the services and government agencies.

Affiliate Relations conducted SEI Visitors Day on February 21, 1991; 46 representatives from industry, government, and academic organizations attended. Additional meetings were conducted with visitors from AT&T, the Air Force, the Navy, Sematech, the National Defense University, and the Logistics Management Institute. The next Visitors Day is scheduled for June 20, 1991. Visitors must register for Visitors Day; on-site registrations will not be accepted.



For More Information

To order an SEI publication, send a written request accompanied by a preaddressed mailing label to:

Software Engineering Institute
ATTN: Publications Requests
Carnegie Mellon University
Pittsburgh, PA 15213-3890

.....

**How to Order
SEI Publications**

To be added to the Bridge mailing list, write to:

Software Engineering Institute
ATTN: Bridge Mailing List
Carnegie Mellon University
Pittsburgh, PA 15213-3890

.....

**How to Get the
SEI Magazine**

For information on opportunities for affiliation, contact:

Software Engineering Institute
ATTN: Mark Coticchia
Carnegie Mellon University
Pittsburgh, PA 15213-3890
(412) 268-6138
Internet: mec@sei.cmu.edu

.....

**How to Become
an SEI Affiliate**

For further information about the SEI, contact:

Message Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890
(412) 268-7700

.....

**How to Get
Additional Information**

